Enhancing Power Wheelchair Mobility with a Cost-Effective Vision-Only Deep Learning System for Obstacle Detection and Avoidance

A. H. Abdul Hafez [*] **Yahya Tawil** [**]

[*] *Department of Computer Science, College of Computer Science and Information Technology, King Faisal University, Al-Ahsa, Saudi Arabia.*
The corresponding author E-mail: aabdulhafiz@kfu.edu.sa
[**] *Department of Computer Engineering, Damascus University, Damascus, Syria*

Abstract: As the demand for power wheelchair (PWC) users increases, obstacle avoidance remains a significant challenge—often beyond the capabilities of manual control. A vision-only autonomous obstacle detection and avoidance system for PWCs is introduced, using just a single monocular camera. This low-cost solution integrates deep learning, computer vision, and mobile robotics without relying on expensive depth sensors. By adapting a deep learning model via transfer learning on a newly developed sidewalk dataset, the system effectively detects obstacles. Once detected, a novel image-space avoidance method dynamically adjusts the wheelchair's motor speeds to navigate safely around obstacles. Deployed on a standard PWC, the system demonstrates object detection accuracy comparable to conventional notebook-based methods while operating at 5 Frames Per Second (FPS) on a Raspberry Pi 4, with a total system cost under $300. Real-world tests on a university campus confirm successful detection and avoidance of nine common sidewalk obstacle classes, with the lightweight design using minimal computational resources. This shows that effective obstacle avoidance can be achieved with minimal hardware, avoiding bulky and costly sensor arrays. The proposed system offers a novel, affordable approach for retrofitting any PWC with minimal modifications using off-the-shelf components. This innovation enhances mobility for PWC users both indoors and outdoors, setting the stage for fully autonomous, vision-only PWCs in the future.

Keywords: Autonomous power wheelchair; assistive technology; computer vision; deep-learning; obstacle detection and avoidance

1. INTRODUCTION

There is an increasing global demand for assistive devices, particularly power wheelchairs (PWCs), which are essential for enhancing users' independence and quality of life (Doss et al., 2023; Organization, 2015; Mortenson et al., 2012). The mobility of PWC users is vital to their independence and quality of life (Edwards and McCluskey, 2010; Frank et al., 2010). Despite their importance, PWC users often face significant challenges navigating obstacle-prone environments such as crowded sidewalks, narrow doorways, and unpredictable outdoor settings, where manual control can be insufficient or unsafe (Torkia et al., 2015; Maule et al., 2022).

Recent review articles highlight the wide range of research efforts dedicated to autonomous wheelchair navigation (Mole et al. (2024); Atoyebi et al. (2025)). A portion of this literature focuses on indoor environments, particularly corridor navigation, where image–based visual servoing (VS) approaches have demonstrated promising performance (Chaumette and Hutchinson (2006); Abdul Hafez (2014); Abdul Hafez et al. (2015)).

However, research addressing outdoor navigation for wheelchairs remains relatively limited. Few existing studies often rely on expensive sensing modalities or computationally intensive architectures, such as LiDAR–driven SLAM systems (Wen et al. (2022a); Panah et al. (2021)), map–based planning strategies (Sorokin et al. (2022)), inverse reinforcement learning frameworks (Wang et al. (2023); Zhang et al. (2023)), or large-scale goal-conditioned vision models (Shah et al. (2023)).

Vision-only solutions reported in the literature typically integrate semantic segmentation with cost-map generation (Wen et al. (2022b)), or address narrower problems such as occlusion-aware local planning (Seo and Kang (2023)), sidewalk navigation (Hafez et al. (2025)), and obstacle avoidance (Tawil and Hafez (2022)). Consequently, a computationally efficient *camera-only* framework specif-

[⋆] This work was supported by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia, under Grant [KFU262092].

ically designed for wheelchair navigation is still lacking. The present work aims to address this gap.

Historically, smart PWCs have been researched extensively since the 1980s, with numerous prototypes developed over decades that explore sensor integration, design, and functionality (Sidik et al., 2018; Leaman and La, 2017; Simpson, 2005; Desai et al., 2017). However, many existing systems rely on expensive sensor suites such as LiDARs and RGB-D cameras (Burhanpurkar et al., 2017; Day et al., 2019), increasing costs and complexity, which limits accessibility for a broader user base. This high cost and complexity motivate the need for a more affordable and reliable solution for autonomous obstacle detection and avoidance.

We now focus on obstacle detection and avoidance in sidewalk and outdoor environments to facilitate mobility for the elderly and people with disabilities using a vision-only system. This work introduces a novel vision-only deep learning approach tailored for PWCs. By leveraging a single low-cost camera and advanced deep learning techniques, the proposed system detects and avoids obstacles in real-time without the need for costly depth sensors. This approach simplifies installation and deployment, reducing overall costs and making advanced autonomous navigation more accessible. Ultimately, the main goal is to pave the way toward fully autonomous, cost-efficient PWCs that are practical for widespread adoption (Morbidi et al., 2022).

The key contributions of this paper are:

(1) **A novel integration framework** combining lightweight deep learning with image-space visual servoing, specifically tailored for wheelchair navigation on embedded hardware;
(2) **A reformulation of IBVS control that accounts for non-holonomic wheelchair constraints** and generates safe avoidance commands directly from 2D image features, eliminating the need for 3D reconstruction;
(3) **The Wheelchair Obstacle Detection Dataset (WODD) dataset,** providing a standardized benchmark for sidewalk obstacle detection to enable reproducible research; and
(4) **A design methodology for cost-constrained assistive robotics** that systematically balances accuracy, speed, and affordability.

The remainder of this paper reviews related work (Section 2), outlines the system architecture (Section 3), details the deep learning model and dataset creation (Section 4), and presents the image-space obstacle avoidance strategy (Section 5). Then, the experimental setup and results are described in (Section 6), followed by discussions on implications, limitations, and future directions (Sections 7 and 8), before concluding with a summary of the key contributions.

2. RELATED WORK

2.1 Obstacle Detection and Avoidance in PWCs

Over the years, various methods for obstacle detection and avoidance in PWCs have been proposed. Early approaches often relied on sensor-heavy systems, such as those by Diao et al. (2017) and Petry et al. (2010), which employed multiple ultrasonic sensors and artificial potential fields for obstacle avoidance. Although pioneering, these methods suffered from reliability issues compared to vision-based solutions. Similarly, Özcelikörs et al. (2014) utilized Kinect sensors to extract depth information for obstacle avoidance, but such depth sensors significantly increase system cost and complexity compared to Raspberry Pi camera used in this work.

Table 1 summarizes the related past works, and focuses on past research and commercial works that use vision sensors for PWC obstacle detection functionality. From the Table, it can be concluded that object detection using computer vision is not yet adopted in the commercial PWC products. Moreover, previous researches presented costly solutions using industrial-type sensors or require a notebook to run the system code. Alternatively, they use affordable sensors but use less reliable or limited techniques. This both demonstrates the “cheap or robust” dilemma and highlights the importance and novelty of this work.

Classical computer vision approaches have also been explored (Kim, 2016; Lee et al., 2017), yet they often suffer from limited performance, unreliable detection under varying conditions, and a dependence on external computing resources that hinder portability. Studies like those by Li et al. (2017) and Burhanpurkar et al. (2017) integrated obstacle avoidance with localization but relied on costly RGB-D sensors and targeted indoor environments. This body of work highlights a dichotomy: high-reliability systems use expensive, heavy sensors, while affordable systems often lack robustness or real-time capabilities.

2.2 Deep Learning and Datasets for Obstacle Detection

Recent advances in deep learning have revolutionized object detection and classification tasks, offering improved accuracy and flexibility over classical methods (O’Mahony et al., 2019). Convolutional Neural Networks (CNNs) facilitate robust feature extraction, and models like YOLO (Lee and Kim, 2020) and SSD (Liu et al., 2016) provide real-time detection capabilities suitable for embedded systems. Prior research has applied deep learning to PWCs for specialized tasks, such as pedestrian detection (Day et al., 2019) and bus-boarding assistance (Ali et al., 2018), but these approaches were limited by the need for high-cost sensors, narrow application scopes, or fragmented integration with control systems.

While several datasets exist for sidewalk navigation and obstacle detection (Park et al., 2020; Ahmed and Yeasin, 2017; Sun et al., 2019; Neuhold et al., 2017), many are proprietary, lack comprehensive annotations, or are unsuitable for PWC applications due to limited scope or size. This scarcity of accessible, relevant data has driven the development of specialized datasets like the presented WODD, tailored for training deep learning models in sidewalk environments.

Table 2 summarizes the number of images, number of classes and annotation type in each dataset. These datasets cannot be effectively used in other projects, as all the datasets mentioned in the table, other than Vistas and SOID, are proprietary without open access, which led us to

Table 1. A Review of research and commercial PWCs with obstacle detection functionality.

Type	Reference	Hardware Setup	Estimated Cost	Main contribution (+) limitations (-)
Research	Kim, E. Y. Kim (2016)	• 1 x CCD Dragonfly camera • 1 x NI DAC USB-6009 • 8 x Ultrasonic sensor	$700	+ Anti collision by combining ultrasonic and computer vision. + Road with sidewalk intersection recognition. – Assistive system. Not autonomous. – No control law provided; system gives fixed commands (Left, Right, straight, Back). – Lack of deployment on SBC or embedded platform.
Research	Lee et al. (2017)	• 1 x Logitech webcam • 2 x Compass module	$70	+ Low-cost. – Primitive thresholding-based image processing. – Lack of deployment on SBC or embedded platform.
Research	Burhanpurkar et al. (2017)	• 1 x Microsoft kinect V2 • 1 x Motor encoder • 1 x On-board computer	$2000	+ PWC Navigation with obstacle avoidance. – High-cost.
Commercial	WHILL Whill (2022)	• 2 x Stereo camera sensors • 1 x Customer interface tablet • 1 x On-board Computer	$4000	+ Navigation using preinstalled maps. + Remote Real-time monitoring through dashboard. – Obstacle collision preventing without avoidance functionality. – Not portable.
Research	Maule, L. Maule et al. (2021)	• 1 x Microsoft Kinect V2 • 1 x Motor encoder • 1 x On-board computer	$2000	+ Hybrid driving options (eye tracking & semiautonomous). – High-cost. – Lack of deployment on SBC or embedded platform.
Research	Day et al. (2019)	• 1 x Raspberry Pi 3 • 1 x Raspberry Pi camera • HC-SR04 ultrasonic Sensors	$200	+ Running using affordable SBC (Raspberry Pi). + Cost efficient. – No control law provided; system gives fixed commands (stop and reverse). – Deep learning and computer vision methods are not integrated with PWC control. – Destructively reused the PWC's Joystick, making it undeployable by normal users.

Table 2. The different datasets built for or used in object detection for robots that use side-walk environments.

Dataset	Size	Annotation	Classes
SideGuide	350,000	Bounding box and semantic polygon	29
SOID[1]	50,000	X	5
Vistas	25,000	Semantic polygon	37
PESID[2]	1,900	Bounding box	10
CDBV[3]	13,500	Bounding box	6

[1] Sidewalk Obstacle Image Dataset (SOID). [2] Pedestrian Safety Application Dataset (PESID).
[3]Chinese Driving from a Bike View (CDBV).

build own dataset. Nine common obstacles usually found on sidewalks are used: pothole, potted plant, fence, traffic sign, trash can, fire hydrant, person, rock, and trunk. The instances are annotated with bounding boxes in the image. The dataset is open-access with no restriction policies.

2.3 Recent Advances in Embedded Vision for Assistive Robotics

Recent years have witnessed continued advancement in efficient deep learning architectures suitable for embedded deployment. Mehta and Rastegari (2022a) introduced MobileViT, demonstrating that vision transformers can be adapted for mobile vision tasks by combining the strengths of CNNs and transformers, achieving competitive accuracy with fewer parameters than traditional CNNs. Subsequent work on MobileViTv2 introduced separable self-attention with linear complexity, further improving efficiency for resource-constrained devices (Mehta and Rastegari, 2022b). Wadekar and Chaurasia (2022) proposed MobileViTv3 with enhanced feature fusion mechanisms, showing the rapid evolution of this research area.

For assistive robotics specifically, recent work has explored various approaches to autonomous navigation. Afonso and Ferreira (2024) demonstrated the application of Deep Deterministic Policy Gradient combined with computer vision for autonomous wheelchair navigation in human-shared environments, achieving improved performance over traditional reinforcement learning approaches. Grigioni et al. (2024) introduced a multi-sensor fusion approach for safe road-crossing by autonomous wheelchairs, along with a novel dataset for this application domain. Poirier et al. (2023) demonstrated the effectiveness of lightweight deep learning models (MobileNet2 with self-attention) for assistive robot control, supporting the viability of such architectures on embedded platforms.

These recent contributions collectively highlight the growing research interest in assistive mobility solutions and the continued relevance of lightweight deep learning approaches for resource-constrained deployment—a gap that the present work addresses through a complete vision-only detection and avoidance system optimized for real-world power wheelchair retrofitting.

2.4 Gaps and Novelty

The literature reveals a prevalent “cheap or robust” trade-off, where existing systems rarely combine low cost with reliable performance. This work addresses this gap by introducing a vision-only, deep learning-based obstacle detection and avoidance system that operates on affordable hardware without sacrificing accuracy or responsiveness. By integrating a lightweight CNN model, the newly developed WODD dataset, and a novel image-space avoidance controller, the proposed approach advances PWC autonomy in a cost-effective manner, offering a practical solution for navigating obstacle-rich environments.

3. SYSTEM OVERVIEW

The proposed system architecture is designed to enable autonomous obstacle detection and avoidance for power wheelchairs using a single low-cost camera and a lightweight computing platform such as the Raspberry Pi. As illustrated in Figure 1, the system integrates hardware components (camera, motor controller, Raspberry Pi) and software modules to achieve real-time obstacle detection and navigation.

The proposed architecture is organized into two primary modules. The first is the Obstacle Detection module, which leverages a fine-tuned deep learning model based on the MobileNetV2 SSD architecture. This model is trained on custom WODD, enabling the system to detect and classify common sidewalk obstacles in real time. The detection module processes live camera feeds, identifies hazards, and outputs bounding boxes with class labels for each detected obstacle.

The second module is the Image-Space Avoidance Controller. Upon receiving obstacle information from the detection module, this controller employs an image-based visual servoing approach to compute linear and angular velocity commands. These commands adjust the wheelchair’s trajectory to push detected obstacles out of a predefined safety margin in the camera’s field of view, thus avoiding collisions. By operating directly on image-space features, the controller eliminates the need for complex 3D reconstruction and reduces computational overhead. Figure 2 shows a flowchart of the proposed algorithm with the avoidance steps, which are followed by the system when the obstacle is inside the Area of Interest (AOI). Each of the major steps in the flow chart is discussed in the following sections.

The proposed design adheres to key criteria: *affordability*, *deployability*, *portability*, and *expandability*. By using off-the-shelf components, minimizing installation complexity, and ensuring lightweight performance, the system is accessible and easily adaptable for future enhancements, making advanced obstacle avoidance technology more practical for widespread PWC use.

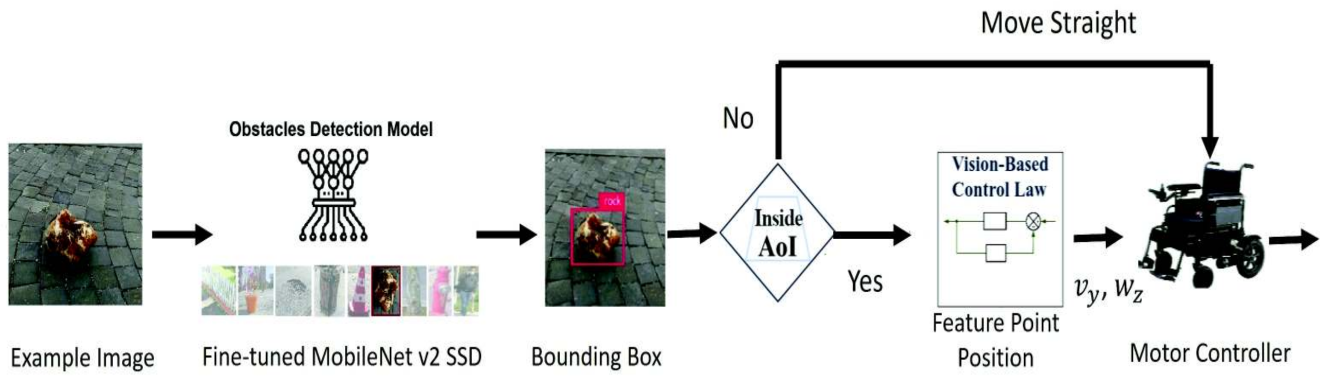

Fig. 1. An illustration of the developed system blocks.

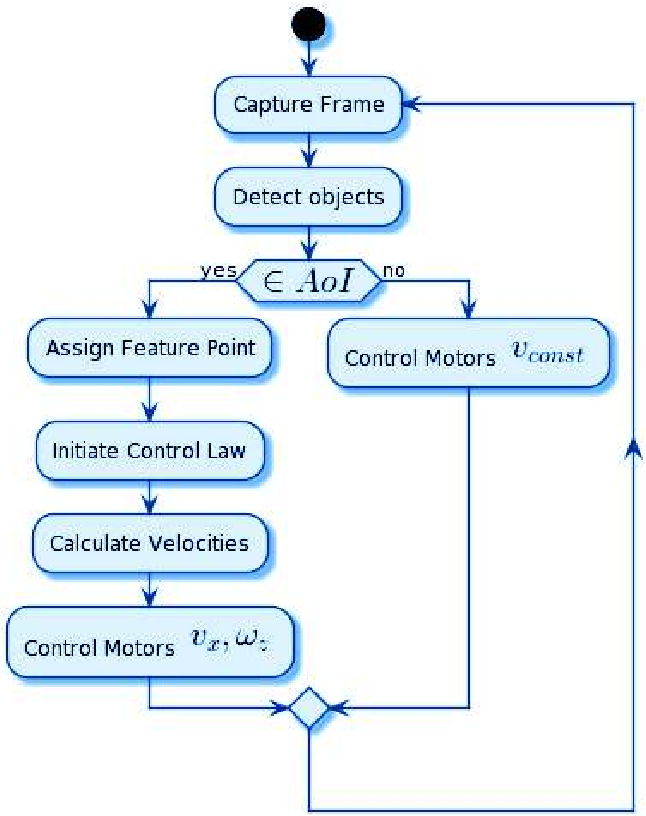

Fig. 2. A flowchart showing the main tasks in the system. Each major step in the flowchart is elaborated upon in its respective sub-section.

4. DEEP LEARNING MODEL AND DATASET

4.1 Model Selection

For real-time obstacle detection on resource-constrained platforms such as the Raspberry Pi, the Single Shot MultiBox Detector (SSD) architecture is used (Sandler et al., 2018; Liu et al., 2016; Lee and Kim, 2020), which is designed for efficiency, combining a lightweight feature extractor with a single-shot detection framework that processes images in one forward pass, as shown in Figure 3.

This choice is justified by the model's low computational overhead and memory footprint, which are critical for embedded devices. Its design balances speed and accuracy, providing sufficient detection performance while maintaining real-time processing capabilities (around 5 FPS) on limited hardware. Additionally, MobileNetV2 SSD supports transfer learning (Sandler et al., 2018), allowing us to fine-tune the model with custom dataset, which further enhances its accuracy in detecting sidewalk obstacles without significantly increasing resource consumption.

4.2 WODD Dataset Creation

To adapt MobileNetV2 SSD for sidewalk obstacle detection, the WODD is created. Data collection involved capturing video footage from a mobile camera mounted on a power wheelchair across diverse outdoor environments, supplemented by web-scraped images from platforms like Google Images and Flickr. Nine common sidewalk obstacles are used: potholes, potted plants, fences, traffic cones, trash cans, fire hydrants, persons, rocks, and trunks. Each image was annotated with bounding boxes corresponding to these obstacle classes.

Various image augmentation techniques are applied to simulate different lighting conditions and sensor noise, thereby increasing dataset diversity and improving model robustness:

- Contrast adjustment: The contrast was modified for each image channel independently. For each pixel value r in a channel with mean m_r, the adjusted value is calculated as $(r - m_r) \times f_c + m_r$, where $f_c = 1.5$ was used to enhance contrast, making obstacles more distinguishable from backgrounds.
- Gamma transformation: This non-linear adjustment was applied to simulate varying illumination conditions. Each pixel value r was transformed to $C \times r^\gamma$, where $\gamma = 1.5$ and $C = 0.9$ were selected to slightly darken images collected on sunny days, simulating overcast conditions or evening light.
- Gaussian noise addition: Random noise sampled from a Gaussian distribution with mean $\mu = 0.0$ and $\sigma = 0.05$ was added to pixel values to simulate sensor noise and low-light conditions where image quality degrades.

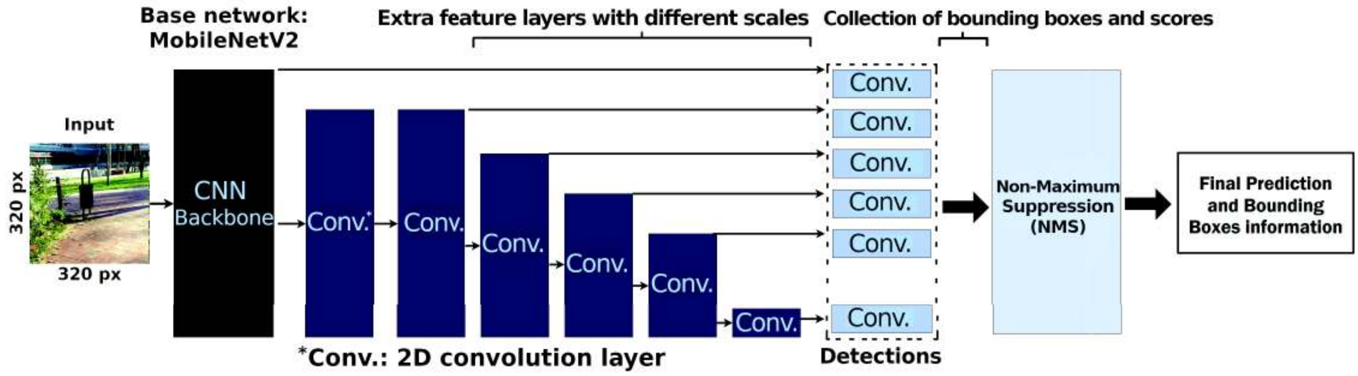

Fig. 3. MobileNetV2 SSD architecture: a single-shot detector combining a lightweight feature extractor backbone (Sandler et al., 2018) with multi-scale prediction and non-maximum suppression (Lee and Kim, 2020).

- Quality adjustment: Images were compressed and decompressed using JPEG encoding with quality factor 12 (on a scale of 1-100, where lower values indicate higher compression). This simulates the reduced image quality that may be encountered when using the lower-cost Raspberry Pi Camera Module compared to the higher-quality camera used for initial data collection.

These augmentations were applied during training in real-time, effectively multiplying the effective dataset size and exposing the model to a wider range of visual conditions, thereby improving its generalization capability. Figure 4 presents a set of sample images showing the different types of the applied augmentation. The WODD dataset supports transfer learning by providing domain-specific examples that fine-tune the pre-trained MobileNetV2 SSD model on obstacle detection tasks relevant to sidewalk navigation, the distribution of data over classes is depicted in Figure 5, thereby enhancing detection accuracy in real-world scenarios.

The WODD dataset consists of 1,411 images annotated with bounding boxes (see Figure 6). At 1.0 GB, the dataset is publicly available [1] and organized into training and testing folders. Each folder contains a JSON file detailing each image's filename, label, and bounding box coordinates (min/max x/y in pixels). Images are 1080x1080 pixels, which simplifies scaling to the network's 320x320 input using a uniform division factor.

Due to the natural frequency of objects—traffic cones being more common than fire hydrants—the dataset is imbalanced. Such imbalance can bias machine-learning models, leading to poorer predictions for less frequent classes. To mitigate this, resampling strategies are applied during training, including undersampling dominant classes and oversampling minority ones through data augmentation (Oksuz et al., 2020). Previous studies, such as (Velasco et al., 2019), have shown that such techniques improve classification results. In experiments, the achieved accuracy was acceptable, with tolerance for correctly detecting the appropriate obstacle classes despite dataset imbalance.

[1] `https://bit.ly/3BPR06U`

4.3 Training & Transfer Learning

The training process began with the MobileNetV2 SSD model pre-trained on the COCO dataset (Common Objects in Context) (Lin et al., 2014), a large-scale object detection dataset containing 80 common object categories with over 200,000 labeled images. Using transfer learning, the model with WODD dataset is fine-tuned, adjusting the classification and regression heads to recognize the nine obstacle classes. Given the class imbalance in WODD—where some obstacles appear more frequently than others— resampling strategies are employed during training. Techniques such as oversampling minority classes through data augmentation and undersampling dominant classes were considered to mitigate bias and improve detection performance across all obstacle types.

The model for Raspberry Pi deployment is optimized by converting it to a TensorFlow Lite format, reducing model size and inference latency. Quantization and other TensorFlow Lite optimizations were applied to ensure low memory usage and fast execution without compromising detection accuracy. The fine-tuned model achieved a mean Average Precision (mAP) of approximately 61%, indicating robust obstacle detection capabilities. This performance, combined with the model's lightweight nature, confirms its suitability for real-time PWC obstacle detection and avoidance tasks on resource-limited hardware.

5. OBSTACLE AVOIDANCE CONTROLLER

5.1 Conceptual Overview

The developed obstacle avoidance strategy is based on image-space visual servoing, which uses visual feedback directly to control the wheelchair's motion. The system defines an *Area of Interest* (AOI) in front of the wheelchair, representing a rectangular region corresponding to a 1 m width and 2 m depth in the real world. Within the AOI, a *safety margin* is established along the image boundaries to ensure obstacles are kept at a safe distance. The problem formulation in the image space is illustrated in Figure 7. The dashed lines here indicate the safety margin of width m (in pixels) from the image boundaries—obstacles should be kept outside this region to ensure safe clearance. The light blue dots represent the obstacle's feature point

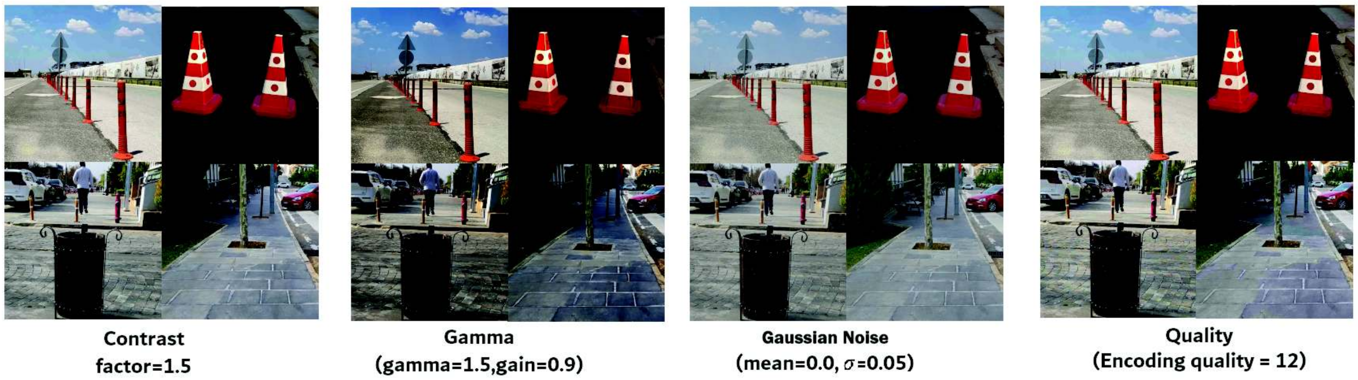

Fig. 4. Sample images from WODD showing the four augmentation effects. They are from left to right: Contrast adjustment with $f_c = 1.5$, Gamma effect transformation with $\gamma = 1.5, C = 0.9$, Gaussian noise with $\mu = 0.0, \sigma = 0.05$, and at the most right is Quality adjustment with encoding quality=12.

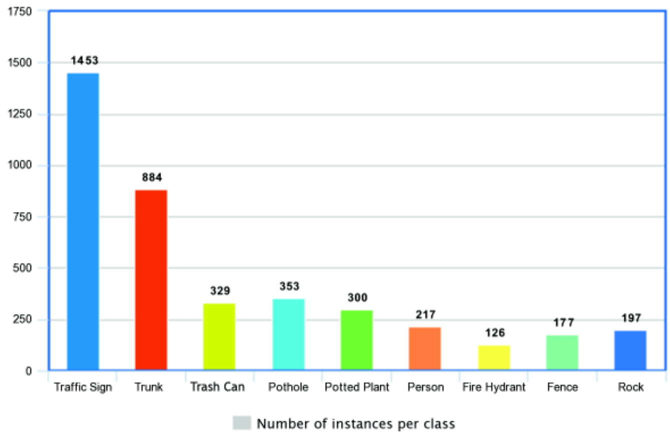

Fig. 5. A bar chart showing the distribution of the 3976 instances over classes in WODD dataset. The traffic cone and trunks classes are generally common among places. This explains the relatively larger number of instances in these two classes.

trajectory during the avoidance maneuver, showing how the control law progressively moves the obstacle toward the safety margin.

When the deep learning module detects an obstacle within the AOI, it outputs a bounding box around the object. The controller uses the center of this bounding box as a feature point in the image plane. If this point lies within the AOI and approaches the safety margin, the avoidance behavior is triggered. The controller then computes motion commands to reposition the obstacle's image towards the safety margin, effectively steering the wheelchair away from potential collisions while resuming its primary trajectory once clear.

5.2 Control Law Derivation

The avoidance mechanism relies on an Image-Based Visual Servoing (IBVS) controller ((Chaumette and Hutchinson, 2006; Abdul Hafez and Jawahar, 2006; Hutchinson and Chaumette, 2007)). Let (u, v) represent the current image coordinates of the obstacle's center, and (u^*, v^*) be the target coordinates at the safety margin. The controller defines an error vector:

$$e = \begin{pmatrix} u - u^* \\ v - v^* \end{pmatrix}, \tag{1}$$

which it strives to minimize.

Using IBVS principles, the relationship between the rate of change of image features and the wheelchair velocities $\dot{r} = (\nu_y, \omega_z)^T$ is given by:

$$\dot{s} = J\dot{r}, \tag{2}$$

where J is the image Jacobian matrix (Chaumette and Hutchinson, 2006; Hutchinson and Chaumette, 2007) . By selecting an exponential convergence strategy, $\dot{e} = -\alpha e$ is imposed, leading to:

$$\dot{u} = -\alpha(u - u^*), \quad \dot{v} = -\alpha(v - v^*). \tag{3}$$

Substituting the specific structure of J for the system and considering planar motion, the key velocity commands are derived:

$$\omega_z = -\alpha \frac{(u - u^*)}{v}, \quad \nu_y = -\alpha \frac{u(u - u^*)}{v} \frac{Z}{\lambda}. \tag{4}$$

Here, α is a gain factor, Z is the assumed constant depth to the obstacle, and λ is the camera focal length in pixels. Equation (4) succinctly captures how the system adjusts angular and linear velocities based on the obstacle's position in the image: as the obstacle approaches the safety margin, the control law increases rotation and lateral movement to steer the obstacle out of the AOI, achieving avoidance while minimally altering the wheelchair's primary path.

6. IMPLEMENTATION DETAILS

6.1 Hardware Setup

The proposed system is built around a Raspberry Pi 4, chosen for its low cost, portability, and sufficient computational power for real-time obstacle detection and avoidance. The Raspberry Pi 4 interfaces with a Raspberry Pi Camera Module v2, which serves as the sole vision sensor. For motion control, a Sabertooth motor driver is employed to interface between the Raspberry Pi and the power wheelchair's motors. This compact setup is integrated directly onto the existing wheelchair platform with minimal modifications, bypassing the joystick and connecting the motor terminals to the controller. The use of off-the-shelf

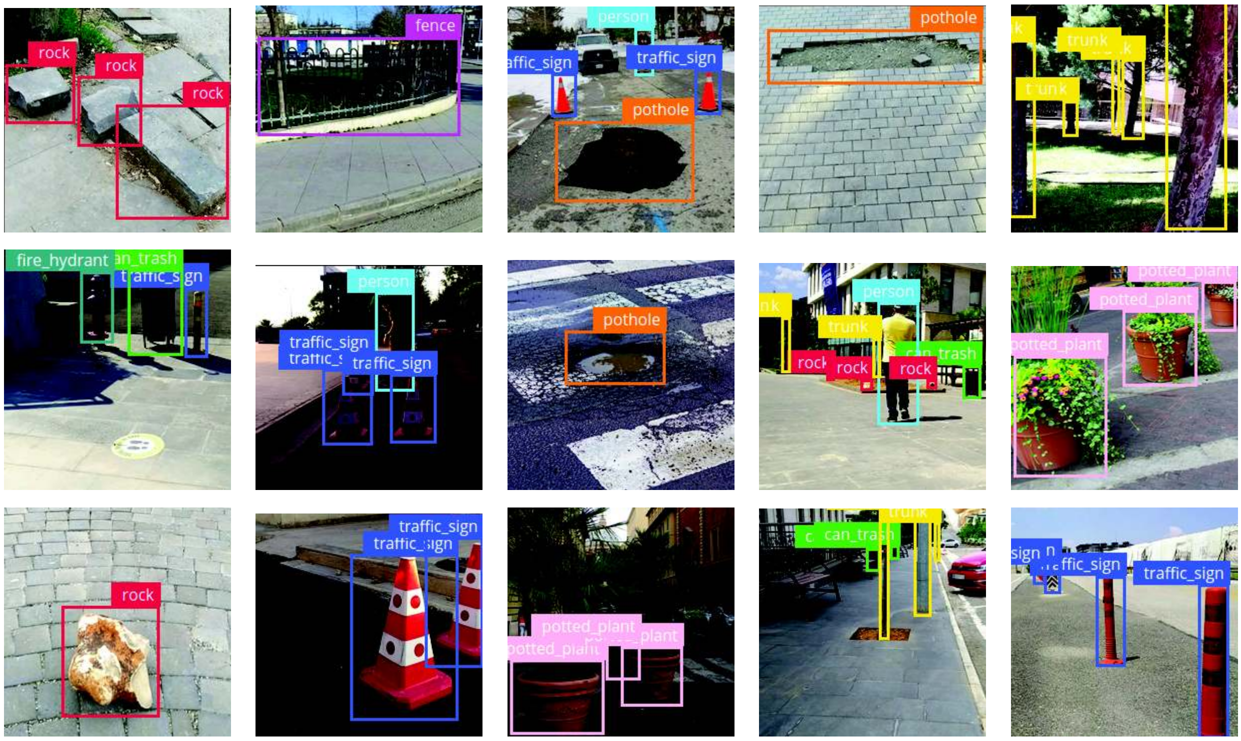

Fig. 6. Samples from the annotated images contained in WODD dataset.

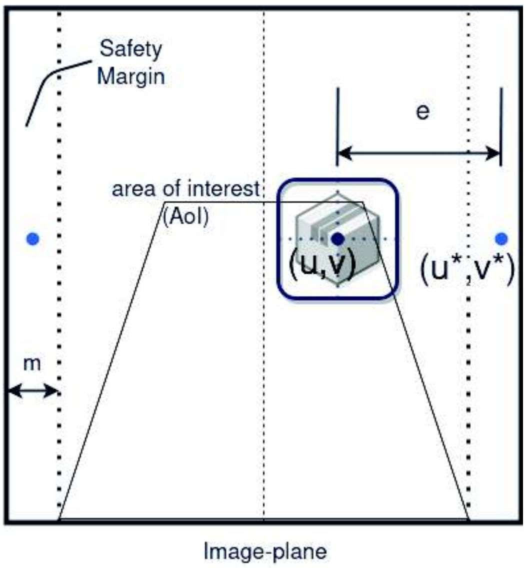

Fig. 7. An illustration of the problem in the image plane. AOI, safety margin, error function e , position of feature point (u, v), and the position of the desired location of feature point (u^*, v^*) are depicted here.

Fig. 8. The PWC used during experiments and development with the installed hardware (battery, Raspberry Pi, motor controller, and camera).

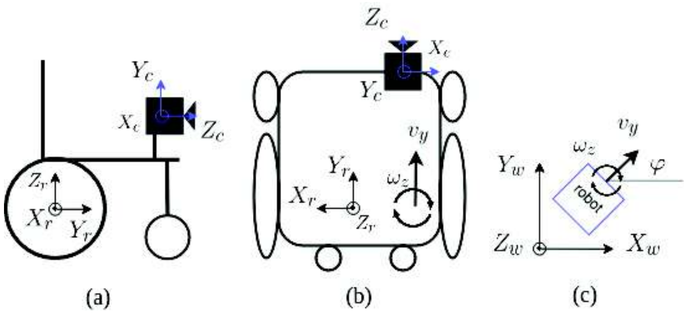

Fig. 9. PWC modelling with camera, robot and world frames (a) Side view of the PWC (b) Top view of the PWC (c) A unicycle model for wheelchair controlled by linear velocity u_y and angular velocity ω_z.

components and a single-camera configuration keeps the total system cost under \$300, emphasizing affordability and portability while ensuring robust performance. Figure 8 shows the hardware setup used during experiments and the development of the system. The PWC and camera configuration are depicted in Figure 9.

6.2 Software & Optimization

The software stack centers on TensorFlow Lite for executing the fine-tuned MobileNetV2 SSD model, selected for its optimized performance on resource-constrained devices. The system runs Python (versions 3.7–3.9 were tested) and leverages TensorFlow Lite 2.8.0 on a 64-bit Raspberry Pi OS, which demonstrated optimal inference speed. Various optimizations were applied to enhance real-time performance:

- Overclocking: The Raspberry Pi's CPU clock was increased beyond the default 700 MHz to 900 MHz, providing faster model inference and control loop execution.
- Process Priority: The Python process running the detection and avoidance algorithms was given higher priority, reducing latency and ensuring more consistent frame rates.
- Power Supply Management: A stable 5V, 3A power source was used to maximize CPU performance and maintain system stability under load.

These optimizations enabled the system to achieve approximately 5 FPS for combined detection and avoidance tasks, with an average CPU usage of around 46% and memory consumption near 150 MB. This balance of speed and resource efficiency confirms the system's suitability for real-time, autonomous obstacle avoidance on a cost-effective and portable hardware platform.

7. EXPERIMENTS AND RESULTS

7.1 Obstacle Detection Evaluation

The performance of fine-tuned MobileNetV2 SSD model on the WODD dataset are evaluated using standard object detection metrics. The model achieved a mean Average Precision (mAP) of approximately 61% on the validation set, demonstrating robust detection across the nine defined obstacle classes. Figure 10 shows the precision-recall curve obtained during validation, indicating a favorable balance between precision and recall at various confidence thresholds.

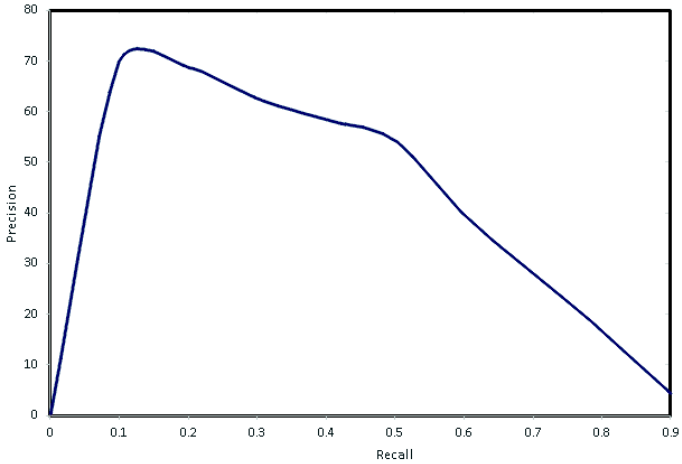

Fig. 10. Precision-recall curve for fine-tuned model.

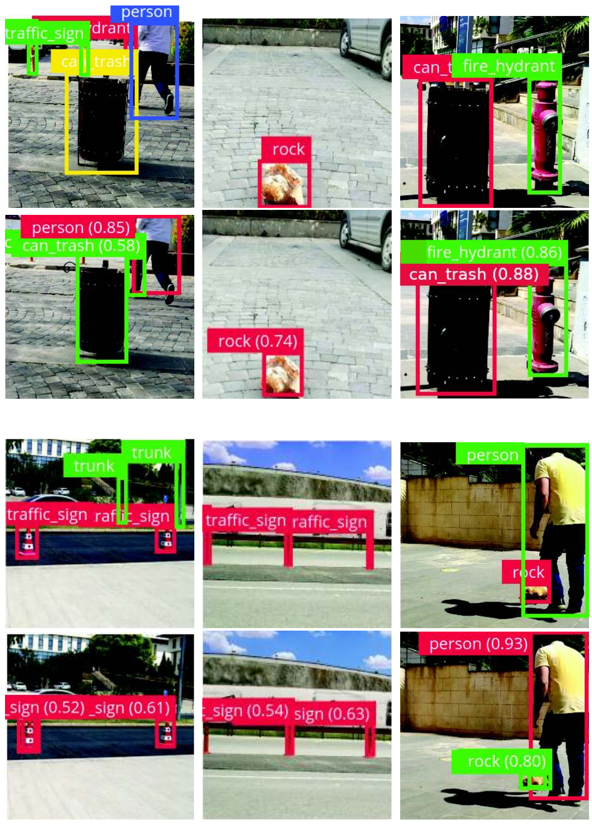

Fig. 11. Samples of the validation images with bounding boxes as ground truth, and under each one the output from the object detection sub-system after fine-tuning using the learning transfer techniques with the developed dataset.

Qualitative results further validate the model's effectiveness. As shown in Figure 11, the detector correctly identifies and classifies obstacles such as traffic cones, persons, and rocks in diverse lighting and environmental conditions. The bounding boxes closely align with ground truth annotations, highlighting the model's accuracy and precision.

When compared to related works, this model's mAP is competitive despite being optimized for real-time performance on a Raspberry Pi. While some sensor-heavy approaches may achieve higher precision, the proposed vision-only system offers a significant reduction in cost and complexity, making it more suitable for widespread use in PWCs.

7.2 Inference Speed Experiments

Inference speed is critical for real-time obstacle avoidance. Experiments under various configurations are conducted to optimize processing speed. Table 3 summarizes the FPS achieved on different hardware and software setups. The best performance, approximately 5.7 FPS, was obtained using a 64-bit Raspberry Pi OS with TensorFlow Lite on Raspberry Pi 4. This setup significantly outperforms alternatives, such as using a 32-bit OS or less optimized models, demonstrating the effectiveness of the software and hardware optimizations. Table 4 shows the CPU and memory usage on Rasbperry Pi 4 while both the detection and avoidance submodules are operating.

The use of TensorFlow Lite, along with overclocking and process prioritization, contributed to the enhanced performance. These optimizations allowed this system to main-

Table 3. MobileNet V2 SSD FPS results on Raspberry Pi 4 & Jetson Nano (CPU[1]) under different setups.

Hardware	Raspberry Pi 4	Raspberry Pi 4	Jetson Nano / CPU	Jetson Nano / CPU
OS [2]	Raspberry Pi OS 64-bit	Raspberry Pi OS 32-bit	Nvidia-jetpack 4.6	Nvidia-jetpack 4.6
Model Format	.tflite model	.tflite model	.PB model	.tflite model
tflite[3] Runtime	2.8.0	2.5.0	2.5.0	2.5.0
Python	3.9	3.7	3.6	3.6
FPS *4	**5.7**	**1.8**	**0.5**	**2.96**

[1] CPU: CPU only without GPU. [2] OS: Operating System. .
[3] tflite: TensorFlow Lite framwork. [4] FPS: Frame Per Second.

Table 4. CPU and memory usage on Rasbperry Pi 4 while both dectection and avoidance submodules are operating.

Process Name	**CPU%**	**Resident Set Size (RSS) / RAM**
model.eim	35%	54.0 MB
python3	9%	105.8 MB

tain a consistent real-time detection and avoidance loop, crucial for safe navigation in dynamic environments.

7.3 Real-World Detection and Avoidance

The presented system is deployed on a power wheelchair and conducted real-world tests on university campus to assess its practical performance. The wheelchair was equipped with the Raspberry Pi 4, camera module, and Sabertooth motor driver, running the integrated detection and avoidance software.

During these experiments, the system successfully detected various obstacles, such as trash cans, traffic cones, and pedestrians, and executed appropriate avoidance maneuvers. Figure 12 displays sample frames where obstacles are detected within the AOI, and the system computes linear and angular velocities to steer clear of collisions. In scenarios where obstacles appeared close to the safety margin, the controller adjusted velocities accordingly—higher angular velocities for closer obstacles and more subtle adjustments when obstacles were farther away.

Trajectory analysis further illustrates the system's behavior. As depicted in Figure 13, the estimated path of the wheelchair deviates smoothly from its initial course to avoid the obstacle. However, the PWC does not fully return to its exact original heading after clearance. This is because the current system is designed specifically for reactive obstacle avoidance rather than complete path planning. The primary objective is to safely navigate around detected obstacles and then resume a straight-line trajectory in the current direction after avoidance, rather than returning to a pre-planned path. In this experiment, the initial heading was approximately 0 degree (straight ahead). After avoiding the obstacle on the right side, the wheelchair continued at a slight leftward offset (approximately 5-10 degrees) from the original heading. This behavior is acceptable for many real-world scenarios where minor path deviations are tolerable, and the user can manually correct the trajectory if needed. For fully autonomous navigation requiring precise path following, a higher-level supervisory system with path planning and replanning capabilities would be necessary—this is identified as a direction for future work.

Overall, the developed system maintained robust performance, adapting to dynamic changes in the environment while processing at an average of 5 FPS. The integration of deep learning-based detection with image-space control allowed for effective real-time obstacle avoidance. These experiments confirm that the proposed approach not only matches theoretical expectations but also performs reliably in practical, real-world conditions, paving the way for broader adoption in assistive mobility technology.

7.4 Comparison with State-of-the-Art Solutions

To contextualize the performance and contributions of the proposed system, Table 5 presents a comparative analysis with existing solutions, including both PWC-specific systems and relevant AGV implementations that address similar navigation challenges. While autonomous obstacle avoidance is not yet widely implemented in commercial PWCs, related technologies have been extensively developed in the AGV and mobile robotics domains. These systems offer valuable benchmarks for comparison, particularly regarding real-time performance, accuracy, and computational efficiency. Key comparison metrics include:

- Detection accuracy (mAP): The proposed system achieves 61% mAP on nine sidewalk obstacle classes. This compares favorably with general-purpose mobile robot detectors (PASCAL VOC reports 64% mAP with similar hardware) while operating on significantly more constrained hardware.
- Inference speed: At 5.7 FPS on Raspberry Pi 4, the proposed system outperforms many embedded vision systems. For comparison, Day et al. (2019) achieved approximately 2-3 FPS on Raspberry Pi 3 with a simpler detector, while AGV implementations using NVIDIA Jetson typically achieve 10-15 FPS but at higher cost ($\geq$\$500) and power consumption ($\geq$10W).
- Cost: At under \$300, the proposed system is among the most affordable autonomous navigation solutions. Commercial AGV systems typically cost \$5,000-\$20,000, while research platforms like the TurtleBot with depth sensors exceed \$1,000.
- Sensor complexity: Unlike systems requiring LiDAR (e.g., Burhanpurkar et al. (2017)) or stereo cameras (e.g., WHILL), the proposed vision-only approach

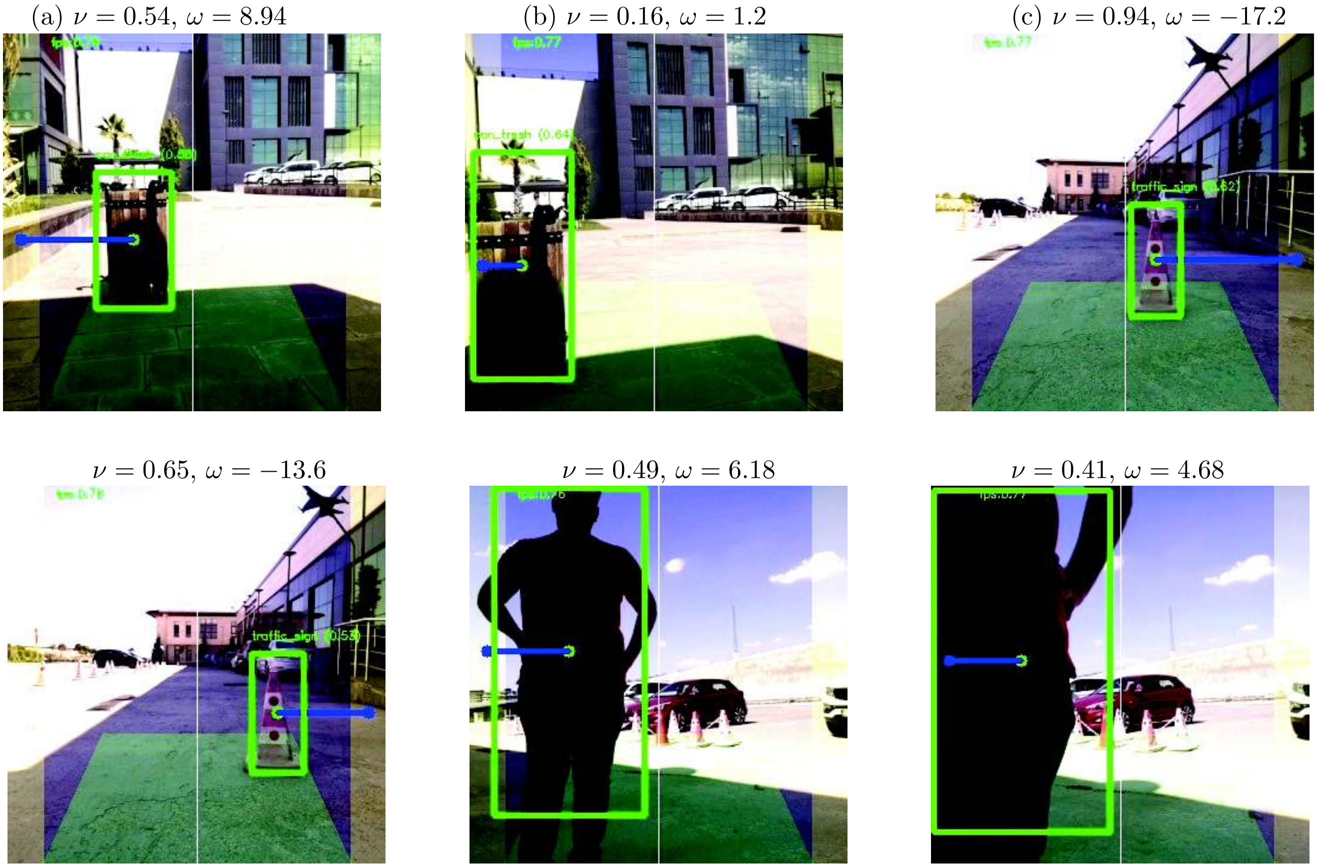

Fig. 12. Different scenes show the detected obstacles (green bounding box) inside AOI (green trapezoid area) and how control law is bringing the feature point (green dot) to the safety margin based on the distance to it (blue line). Under each image the action is generated using the velocities calculated in the control law.

minimizes hardware requirements while maintaining robustness.

- Integration completeness: Unlike partial solutions that address only detection (e.g., Day et al. (2019)) or only control (e.g., Kim (2016)), the proposed system provides end-to-end integration from perception to actuation.

Table 5 summarizes this comparison, demonstrating that the proposed system occupies a unique position in the design space—combining affordability, real-time performance, and complete integration in a way that existing solutions do not achieve.

8. DISCUSSION

8.1 System Expandability

The current architecture of the obstacle detection and avoidance system is inherently expandable. While this work focuses on forward navigation using a single front-mounted camera, the framework can readily extend to bidirectional navigation by incorporating additional camera modules at the rear or sides of the wheelchair. Such a multi-camera setup would allow for continuous monitoring of the environment in all directions, enhancing safety and navigation flexibility. Moreover, the modular design facilitates integration with higher-level supervisory systems that could incorporate advanced decision-making, route planning, or user-specific customization. These supervisory layers could coordinate among multiple sensors and behavioral modules, enabling the wheelchair to adapt to complex environments and user preferences.

8.2 Handling Multiple Obstacles

The current approach simplifies the avoidance task by selecting a single "main obstacle" within the Area of Interest to guide maneuvering. This assumption, while effective in scenarios with isolated hazards, may not scale to environments densely populated with obstacles. Future work should address simultaneous detection and avoidance of multiple obstacles, potentially by prioritizing obstacles based on proximity and collision risk or by segmenting the environment into sub-regions for parallel processing. Incorporating strategies from multi-object tracking and advanced planning algorithms could enable the system to navigate complex, cluttered spaces more effectively.

8.3 Handling Dynamic Obstacles

The current implementation focuses primarily on stationary obstacles, which represent the majority of sidewalk hazards such as trash cans, traffic cones, fire hydrants, and rocks. However, persons (pedestrians) are dynamic obstacles that may change their position and trajectory unpredictably. This requires special consideration in the avoidance strategy. In the present system, when a person

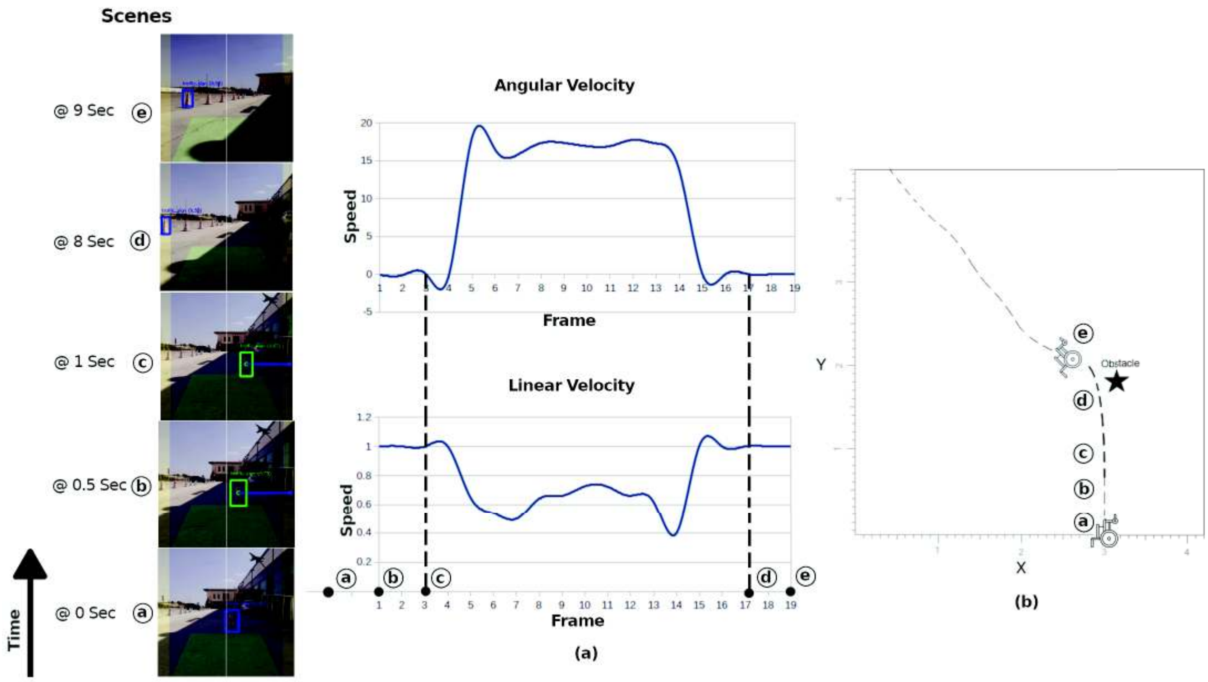

Fig. 13. Experimental results showing obstacle avoidance behavior. (a) Linear velocity and angular velocity computed by the control law over time (frame number). (b) Estimated trajectory path of the PWC during obstacle avoidance, with the obstacle location marked by a red star. The path shows the wheelchair's position over time, with points (a)-(e) corresponding to the following stages: (a) Initial straight-line motion before obstacle detection; (b) Obstacle enters AOI, avoidance maneuver begins with increasing angular velocity; (c) Maximum deviation from original path as obstacle is being pushed to safety margin; (d) Obstacle cleared from AOI, angular velocity decreasing; (e) Return to approximately the original heading, resuming main navigation task.

Table 5. Comparative Analysis with Existing Solutions

System	Platform	Sensors	Cost	FPS	mAP	Integration Level	Limitations
This system	PWC	Monocular camera	≤$300	5.7	61%	Complete (detection + control)	9 classes only
Day et al. (2019)	PWC	Camera + ultrasonic	$200	2-3	N/A	Partial (detection only)	No control law
Kim (2016)	PWC	Camera + ultrasonic	$700	N/A	N/A	Partial (assistive only)	Fixed commands only
Burhanpurkar et al. (2017)	PWC	Kinect + encoder	$2000	N/A	N/A	Complete	High cost
Typical AGV	Industrial	LiDAR + cameras	$5000+	10-15	70-80%	Complete	Cost prohibitive
TurtleBot3	Research	LiDAR + RGB-D	$1000+	5-10	Varies	Complete	Limited battery

is detected within the AOI, the same image-space control law is applied, treating the person's current position as a static feature point at the moment of detection. While this approach has proven effective in the experiments conducted (as shown in Figure 12, where persons were successfully avoided), it does not account for potential sudden movements by the pedestrian. For future implementations, several strategies could be incorporated to better handle dynamic obstacles such as motion prediction, adaptive control gain, behavior modeling, and shared control.

8.4 Limitations & Future Directions

The present system assumes a fixed straight-line navigation path between avoidance maneuvers and focuses on nine specific obstacle classes. Such constraints limit its applicability in more dynamic navigation tasks that require curved paths, turning, or complex route planning. Additionally, the limited set of obstacle classes, while representative of common sidewalk hazards, may not cover all real-world scenarios. Furthermore, the current frame rate is essentially adequate for typical wheelchair operating speeds (0.3-0.8 m/s) as documented in the literature. For higher speeds, either frame rate must increase or operating speed must be reduced during autonomous mode.

Future research will expand the dataset to include a broader range of obstacles and environmental conditions. Further enhancements may involve integrating path planning algorithms to allow for dynamic route adjustments post-avoidance, thereby transitioning from reactive obstacle avoidance to proactive navigation strategies. In addition, future iterations will explore hardware acceleration options including the Coral Edge TPU and Raspberry Pi

AI Kit, which could increase frame rate to 15-20 FPS, enabling higher resolution input and more complex detection models while maintaining real-time performance. This would allow expansion to additional obstacle classes and more sophisticated control strategies.

9. CONCLUSION

This work presents a robust, affordable vision-only obstacle detection and avoidance system for power wheelchairs. By combining a lightweight deep learning model with an image-space visual servoing controller, the proposed approach enables real-time, autonomous navigation using only a single low-cost camera and embedded hardware. The creation of the publicly accessible WODD dataset and the integration of efficient control algorithms mark significant strides toward practical deployment. Practical implications include increased mobility and safety for wheelchair users without the need for expensive sensors, making advanced autonomous features accessible to a wider population. The system's cost-effectiveness and portability suggest strong potential for widespread adoption. Future research will focus on expanding obstacle classes, refining path planning strategies, and integrating higher-level supervisory systems to progress toward fully autonomous, cost-effective power wheelchairs that adapt seamlessly to complex, real-world environments.

ACKNOWLEDGMENT

The initial stage of this work was supported by CARA Syria Program. The Titan Xp used for this research was donated by the NVIDIA Corporation.

REFERENCES

Abdul Hafez, A.H. (2014). Visual servo control by optimizing hybrid objective function with visibility and path constraints. *Journal of Control Engineering and Applied Informatics*, 16(2), 120–129.

Abdul Hafez, A.H. and Jawahar, C.V. (2006). Probabilistic integration of 2d and 3d cues for visual servoing. In *2006 9th International Conference on Control, Automation, Robotics and Vision*, 1–6. IEEE.

Abdul Hafez, A.H., Nelakanti, A.K., and Jawahar, C. (2015). Path planning for visual servoing and navigation using convex optimization. *International journal of robotics and automation*, 30(3), 299–307.

Afonso, P.D.A. and Ferreira, P.R. (2024). Deep deterministic policy gradient and computer vision for autonomous wheelchair navigation in human-shared environments. In *2024 Brazilian Symposium on Robotics (SBR) and 2024 Workshop on Robotics in Education (WRE)*, 103–108. IEEE.

Ahmed, F. and Yeasin, M. (2017). Optimization and evaluation of deep architectures for ambient awareness on a sidewalk. In *2017 International Joint Conference on Neural Networks (IJCNN)*, 2692–2697.

Ali, S., Mamun, S.A., Fukuda, H., Lam, A., Kobayashi, Y., and Kuno, Y. (2018). Smart robotic wheelchair for bus boarding using cnn combined with hough transforms. In *International Conference on Intelligent Computing*, 163–172.

Atoyebi, O., Wister, A., Mattie, J., Beadle, J., Gutman, G., Chaudhury, H., Sparrey, C.J., Jones, O.Y., Mortenson, W.B., O'Dea, E., et al. (2025). Power assist addons for adult manual wheelchair users: A scoping review. *Assistive Technology*, 1–12.

Burhanpurkar, M., Labbé, M., Guan, C., Michaud, F., and Kelly, J. (2017). Cheap or robust? the practical realization of self-driving wheelchair technology. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, 1079–1086.

Chaumette, F. and Hutchinson, S. (2006). Visual servo control i: Basic approaches. *IEEE Robotics Automation Magazine*, 13, 82–90.

Day, C., McEachen, L., Khan, A., Sharma, S., and Masala, G. (2019). Pedestrian recognition and obstacle avoidance for autonomous vehicles using raspberry pi. In *Proceedings of SAI Intelligent Systems Conference*, 51–69.

Desai, S., Mantha, S., and Phalle, V. (2017). Advances in smart wheelchair technology. In *2017 International Conference on Nascent Technologies in Engineering (ICNTE)*, 1–7.

Diao, C., Jia, S., Zhang, G., Sun, Y., Zhang, X., Xue, Y., and Li, X. (2017). Design and realization of a novel obstacle avoidance algorithm for intelligent wheelchair bed using ultrasonic sensor. In *2017 Chinese Automation Congress (CAC)*, 4153–4158.

Doss, A.S.A., Lingampally, P.K., Nguyen, G.M.T., and Schilberg, D. (2023). A comprehensive review of wearable assistive robotic devices used for head and neck rehabilitation. *Results in Engineering*, 19, 101306.

Edwards, K. and McCluskey, A. (2010). A survey of adult power wheelchair and scooter users. *Disability and Rehabilitation: Assistive Technology*, 5, 411–419.

Frank, A., Neophytou, C., Frank, J., and Souza, L. (2010). Electric-powered indoor/outdoor wheelchairs (epiocs): Users' views of influence on family, friends, and carers. *Disability and Rehabilitation: Assistive Technology*, 5, 327–338.

Grigioni, C., Corradini, F., Antonucci, A., Guzzi, J., and Flammini, F. (2024). Safe road-crossing by autonomous wheelchairs: A novel dataset and its evaluation. In *Computer Safety, Reliability, and Security. SAFECOMP 2024 Workshops*, volume 14989 of *Lecture Notes in Computer Science*, 47–60. Springer.

Hafez, A.A., Osman, I.H., Uğur, E., and Kara, T. (2025). Lightweight gaussian process–based visual servoing for autonomous wheelchair sidewalk navigation. *IEEE Access.*

Hutchinson, S. and Chaumette, F. (2007). Visual servo control ii: Advanced approaches. *IEEE Robotics and Automation Magazine*, 14, 109–118.

Kim, E. (2016). Wheelchair navigation system for disabled and elderly people. *Sensors*, 16(11), 1806.

Leaman, J. and La, H. (2017). A comprehensive review of smart wheelchairs: Past, present, and future. *IEEE Transactions on Human-Machine Systems*, 47(4), 486–499.

Lee, Y. and Kim, Y. (2020). Comparison of cnn and yolo for object detection. *Journal of The Semiconductor Display Technology*, 19, 85–92.

Lee, Y., Lim, J., Eu, K., Goh, Y., and Tew, Y. (2017). Real-time image processing based obstacle avoidance and navigation system for autonomous wheelchair application. In *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 380–385.

Li, Z., Xiong, Y., and Zhou, L. (2017). Ros-based indoor autonomous exploration and navigation wheelchair. In *2017 10th International Symposium on Computational Intelligence and Design (ISCID)*, 132–135.

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C.L. (2014). Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 740–755. Springer.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C., and Berg, A. (2016). Ssd: Single shot multibox detector. In *European Conference on Computer Vision*, 21–37.

Maule, L., Luchetti, A., Zanetti, M., Tomasin, P., Pertile, M., Tavernini, M., Guandalini, G., and Cecco, M.D. (2021). Roboeye, an efficient, reliable, and safe semi-autonomous gaze driven wheelchair for domestic use. *Technologies*, 9(16).

Maule, L., Zanetti, M., Luchetti, A., Tomasin, P., Dallapiccola, M., Covre, N., Guandalini, G., and Cecco, M.D. (2022). Wheelchair driving strategies: A comparison between standard joystick and gaze-based control. *Assistive Technology*, 1–13.

Mehta, S. and Rastegari, M. (2022a). Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer. In *International Conference on Learning Representations (ICLR)*. ArXiv:2110.02178.

Mehta, S. and Rastegari, M. (2022b). Separable self-attention for mobile vision transformers. *arXiv preprint*, arXiv:2206.02680.

Mole, A., Gurav, S., Shinde, S., Bhagwat, Y., and Patil, M.B. (2024). Survey on smart wheelchairs. *Journal of Electronics and Telecommunication System Engineering*, 43–50.

Morbidi, F., Devigne, L., Teodorescu, C., Fraudet, B., Leblong, E., Carlson, T., Babel, M., Caron, G., Delmas, S., and Pasteau, F. (2022). Assistive robotic technologies for next-generation smart wheelchairs: Codesign and modularity to improve users' quality of life. *IEEE Robotics and Automation Magazine*, 2–14.

Mortenson, W., Miller, W., Backman, C., and Oliffe, J. (2012). Association between mobility, participation, and wheelchair-related factors in long-term care residents who use wheelchairs as their primary means of mobility. *Journal of The American Geriatrics Society*, 60, 1310–1315.

Neuhold, G., Ollmann, T., Bulo, S.R., and Kontschieder, P. (2017). The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE International Conference on Computer Vision*, 4990–4999.

Oksuz, K., Cam, B., Kalkan, S., and Akbas, E. (2020). Imbalance problems in object detection: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43, 3388–3415.

Organization, W.H. (2015). *World Health Statistics 2015*. World Health Organization.

O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G., Krpalkova, L., Riordan, D., and Walsh, J. (2019). Deep learning vs. traditional computer vision. In *Science and Information Conference*, 128–144.

Panah, A., Motameni, H., and Ebrahimnejad, A. (2021). An efficient computational hybrid filter to the slam problem for an autonomous wheeled mobile robot. *International Journal of Control, Automation and Systems*, 19, 3533–3542.

Park, K., Oh, Y., Ham, S., Joo, K., Kim, H., Kum, H., and Kweon, I. (2020). Sideguide: A large-scale sidewalk dataset for guiding impaired people. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 10022–10029.

Petry, M., Moreira, A., Braga, R., and Reis, L. (2010). Shared control for obstacle avoidance in intelligent wheelchairs. In *2010 IEEE Conference on Robotics, Automation and Mechatronics*, 182–187.

Poirier, S., Côté-Allard, U., Routhier, F., and Campeau-Lecours, A. (2023). Efficient self-attention model for speech recognition-based assistive robots control. *Sensors*, 23(13), 6056.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4510–4520.

Seo, D. and Kang, J. (2023). Collision-avoided tracking control of uav using velocity-adaptive 3d local path planning. *International Journal of Control, Automation and Systems*, 21(1), 231–243.

Shah, D., Sridhar, A., Bhorkar, A., Hirose, N., and Levine, S. (2023). Gnm: A general navigation model to drive any robot. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 7226–7233. IEEE.

Sidik, M., Ghani, S., Ishak, M., Harun, W., and Daud, N. (2018). A review on electric wheelchair innovation to ease mobility and as a rehabilitation tool for spinal cord impairment patients. *Internet Journal on Engineering and Technology*, 10(3), 803–815.

Simpson, R. (2005). Smart wheelchairs: A literature review. *Journal of Rehabilitation Research and Development*, 42(4), 423.

Sorokin, M., Tan, J., Liu, C.K., and Ha, S. (2022). Learning to navigate sidewalks in outdoor environments. *IEEE Robotics and Automation Letters*, 7(2), 3906–3913.

Sun, C., Su, J., Ren, W., and Guan, Y. (2019). Wide-view sidewalk dataset based pedestrian safety application. *IEEE Access*, 7, 151399–151408.

Tawil, Y. and Hafez, A.A. (2022). Deep learning obstacle detection and avoidance for powered wheelchair. In *2022 Innovations in Intelligent Systems and Applications Conference (ASYU)*, 1–6. doi:10.1109/ASYU56188.2022.9925493.

Torkia, C., Reid, D., Korner-Bitensky, N., Kairy, D., Rushton, P., Demers, L., and Archambault, P. (2015). Power wheelchair driving challenges in the community: A users' perspective. *Disability and Rehabilitation: Assistive Technology*, 10(3), 211–215.

Velasco, J., Pascion, C., Alberio, J., Apuang, J., Cruz, J., Gomez, M., Jr, B.M., Tuala, L., Thio-ac, A., and Jr, R.J. (2019). A smartphone-based skin disease classification using mobilenet cnn. *ArXiv Preprint ArXiv:1911.07929*.

Wadekar, S.N. and Chaurasia, A. (2022). Mobilevitv3: Mobile-friendly vision transformer with simple and effective fusion of local, global and input features. *arXiv preprint*, arXiv:2209.15159.

Wang, T., Dhiman, V., and Atanasov, N. (2023). Inverse reinforcement learning for autonomous navigation via differentiable semantic mapping and planning. *Autonomous Robots*, 1–22.

Wen, M., Dai, Y., Chen, T., Zhao, C., Zhang, J., and Wang, D. (2022a). A robust sidewalk navigation method for mobile robots based on sparse semantic point cloud. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 7841–7846. IEEE.

Wen, M., Zhang, J., Chen, T., Peng, G., Chia, T., and Ma, Y. (2022b). Vision based sidewalk navigation for last-mile delivery robot. In *2022 17th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 249–254. IEEE.

Whill, I. (2022). Autonomous solutions. URL `https://whill.inc/gb/autonomous-solutions/`.

Zhang, T., Liu, Z., Pu, Z., Yi, J., Liang, Y., and Zhang, D. (2023). Robot subgoal-guided navigation in dynamic crowded environments with hierarchical deep reinforcement learning. *International Journal of Control, Automation and Systems*, 1–13.

Özcelikörs, M., Coşkun, A., Say, M., Yazici, A., Yayan, U., and Akcakoca, M. (2014). Kinect-based intelligent wheelchair navigation with potential fields. In *2014 IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA)*, 330–337.